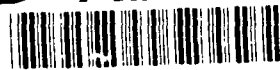AD-A238 710

91-05812

PROGRAM TO PRODUCE TABULATED DATA SET DESCRIBING NSWC BURN MODEL FOR HYDRODYNAMIC COMPUTATIONS

BY LEWIS C. HUDSON III and KIBONG KIM

RESEARCH AND TECHNOLOGY DEPARTMENT

11 SEPTEMBER 1990

DTIC
ELECTE
JUL 29 1991
S B D

NAVAL SURFACE WARFARE CENTER

Dahlgren, Virginia 22448-5000 ● Silver Spring, Maryland 20903-5000

FOREWORD

The research effort documented in this report was
performed under the Advanced Technology Torpedo Thrust
Program and funded by Office of Naval Technology. A general
purpose program has been developed which applies the concepts
of the NSWC Burn Model and generates a tabulated data set.
From this tabulated data set, any one-, two-, or three-
dimensional hydrocode containing a reactive mixture scheme
can determine a new burn fraction based on a previous burn
fraction, pressure, and time step.

The illustrations in this report appear after the body
of the report.

The author wishes to acknowledge the helpful insights of
Dr. Raafat Guirguis of the Naval Surface Warfare Center on
how the NSWC Burn Model works, and Drs. Schittke and Feisler
of Industrieanlagen-Betriebsgesellschaft for their suggestion
to develop a general purpose program to tabulate the data and
an example of how to proceed with the effort.

Approved by:

WILLIAM H. BOHLI, Acting
Energetic Materials Division

Accession For

NTIS CRA&I	☑
DTIC TAB	☐
Unannounced	☐
Justification	

By
Distribution/

Availability Codes

| | Avail and/or |
| Dist | Special |

A-1

CONTENTS

ILLUSTRATIONS

TABLES

SECTION 1

INTRODUCTION

This report describes the transformation of the NSWC Burn Model, described in Reference 1, into a general purpose program which produces a simple set of information. This information can be transformed into a tabular database that can be used by any one-, two-, or three-dimensional hydrocode which contains a reactive mixture scheme. The term, general purpose program, is used throughout this report to emphasis the fact that the computer program which produces the tabulated data set is separate from the hydrocode run which uses the tabulated data set.

The basic equations, described in Reference 1, which present the concepts of the NSWC Burn Model are solved explicitly by a general purpose program in a Lagrangian coordinate system. These equations and the solution method are described in Section 2. Relationships which prescribe the necessary parameters for a hydrocode computation are shown in Section 3. Section 4 presents some conclusions and direction of future work.

SECTION 2

HOW THE NSWC BURN MODEL
IS USED IN A GENERAL PURPOSE PROGRAM

The NSWC Burn Model assumes that a typical pore and
binder in a high explosive can be modeled by a void region
inside a sphere. Therefore, it is referred to as a hollow
sphere model. The size of the hollow portion of the sphere
representing the void and binder is in proportion with the
global percentages of the component explosive, binder, and
pore. The solid portion of the hollow sphere located between
r_i, the minimum radius of the solid portion of the hollow
sphere, and r_o, the maximum radius of the solid portion of the
hollow sphere, is composed of the component explosive.
Figure 1-a shows a typical situation in a real component
explosive. Figure 1-b shows how the NSWC Burn Model views a
typical situation in a real component explosive. Figure 1-c
shows the region of computation of the NSWC Burn Model.

The NSWC Burn Model divides the portion of the sphere
between r_i and r_o into a number of one-dimensional subcells
which can be equally or geometrically spaced. In general, it
seems that geometric spacing is more advantageous since the
region requiring the greatest resolution is near r_i. This is
due to the fact that the NSWC Burn Model is only used to
determine the global burn fraction until a certain criteria
is reached, indicating that surface burning has become
predominant in determining the burn fraction. Then the
global burn fraction can be determined according to a surface
burn model.

It is assumed in the general purpose program presented here that the temperature throughout the hollow portion of the sphere representing the void and binder is constant. The value used for this region is equal to the value of the temperature at the location r_i initially, and at the location corresponding to the inner surface of the solid portion of the sphere as it moves into the solid region due to the conversion of solid subcells into burned gas products as time progresses. As time progresses, the small portion of burned gas products over all the subcells is collected in the hollow portion of the sphere in a cumulative manner.

In the general purpose program describing the NSWC Burn Model, each subcell is bounded by two location values, the difference of which is the subcell width. The local burn fraction and temperature are found at the center of each subcell. The velocity is found at the two locations which bound the subcell region. Initially, it is assumed that the local burn fraction is zero, the temperature is the ambient temperature, and the velocity is zero for all subcells.

Initially, a pressure due to the surroundings, p_o, is applied at r_o and the pressure of the gas in the hollow portion, p_g, of the sphere is ambient pressure or some specified condition and applied at r_i. As the pressure, p_o, is applied to the outer surface of the hollow sphere model, it is assumed that there is a global mechanical deformation in progress near the pore, and that the material behavior is such that the stress is dependent on strain and strain rate. This results in an increase in p_g. The processes which occur are described in greater detail in Reference 1. Two assumptions are made, the elastic portion of mechanical deformation can be ignored and the bulk compressibility of the component explosive can be neglected. This greatly simplifies the equations needed to describe the NSWC Burn Model for a hollow sphere.

Also, the location of the subcells in the hollow sphere model moves toward the center of the sphere as the pressure, p_o, is applied to the outer surface of the hollow sphere model. Likewise, as the component explosive begins to burn, the gas pressure, p_g, in the hollow portion of the sphere begins to increase, resulting in a force being applied to the surface located at r_i and directed toward the outer surface of the sphere. This results in additional changes to subcell boundary locations. The equation used to determine the velocity, $v(r,x,t)$, of the boundaries associated with each subcell is the following,[1]

$$v(r,x,t) = \frac{\gamma \left(p_o - p_g - 2\sqrt{3}\ k\ \ln(r_0/r_i)\right)}{2 \left(r_o^{-3} - r_i^{-3}\right) r^2 k} .$$ (1)

The variables used in Equation (1) are described below.

The basic equations used in the general purpose program to describe a hollow sphere model representative of the NSWC Burn Model in which an Arrhenius reaction is applied to the hot spot temperature which is modified by thermal diffusion, are the following,[1]

$$\frac{d\lambda(x,t)}{dt} = \int_{r_i}^{r_0} \frac{\frac{d\Lambda(r,x,t)}{dt}\ 4\ \pi\ r^2}{\frac{4}{3}\ \pi\ \left(r_o^3 - r_i^3\right)}\ dr ,$$ (2)

$$\frac{d\Lambda(r,x,t)}{dt} = (1 - \Lambda(r,x,t))\ Z\ \exp\left(-\frac{T^*}{T(r,x,t)}\right) ,$$ (3)

$$T(r,x,t) = T_0(r,x,t) + \int_0^t \frac{dT}{dt}(r,x,t)dt , \text{ and}$$ (4)

5

$$\frac{dT(r,x,t)}{dt} = \frac{2.25 \; \gamma \left(p_o - p_g - 2\sqrt{3} \; k \; \ln(r_o/r_i) \right)^2}{\rho \; C_p \left(r_i^{-3} - r_o^{-3} \right) r^6 \; k}$$

$$+ \; \frac{1}{\rho \; C_p} \frac{1}{r^2} \frac{\partial}{\partial r} \left(r^2 \; k^* \; \frac{\partial T(r,x,t)}{\partial r} \right) + \frac{Q}{C_p} \frac{d\Lambda(r,x,t)}{dt} \; , \qquad (5)$$

where r is the location of the subcell being evaluated at time, t; $\lambda(x,t)$ is the global burn fraction of the system; $\Lambda(r,x,t)$ is the local burn fraction of an individual subcell in the hollow sphere model; and $T(r,x,t)$ is the temperature of an individual subcell in the hollow sphere model. Equations (3) through (5) are solved explicitly in the Lagrangian subcell coordinate system. The variable x indicates the global computational cell for which the hollow sphere model is being computed. For example, in a three-dimensional Cartesian coordinate system, x would be a function describing a unique X, Y, and Z location for the global computational cell. In the general purpose program described in this paper, the hollow sphere model is evaluated for one computational cell. Therefore, x is set to 1. $T_0(r,x,t)$ is the initial temperature of an individual subcell in the hollow sphere model, Z is the pre-exponential factor in the Arrhenius kinetics, T^* is the activation temperature, ρ is the density, k^* is the thermal conductivity, C_p is the heat capacity, Q is the heat of reaction of the explosive, and γ is a number which affects how fast the global burn fraction rises but not the asymptote to which the global burn fraction approaches. The values for the constants used to model PBX-9404[1] are shown in Table 1 on the next page.

Due to the nature of Equation (4) for dT(r,x,t)/dt, it is necessary to determine the quantity,

$$\frac{1}{\rho \; C_p} \frac{1}{r^2} \frac{\partial}{\partial r} \left(r^2 \; k^* \; \frac{\partial T(r,x,t)}{\partial r} \right) , \qquad (6)$$

6

TABLE 1.   PBX-9404 PARAMETERS USED IN THE NSWC BURN MODEL

Z	5.0×10^{13}	μsec^{-1}
T^*	26500	$^{\circ}K$
T_0	300	$^{\circ}K$
ρ	1.84	g/cm^3
C_p	1.4×10^{-5}	$cm^2/\mu sec^2/^{\circ}K$
k	8.0×10^{-5}	Mbar
Q	5.439×10^{-2}	$cm^2/\mu sec^2$
k^*	8×10^{-14}	$cm/\mu sec/^{\circ}K$
r_i	0.0039	cm
r_o	0.01	cm

for both boundaries of a subcell. This takes into account
both the temperature transferred from the neighboring subcell
on one side of a particular subcell and the temperature
transferred to the neighboring subcell on the other side of a
particular subcell.

The general purpose program which uses Equations (1)
through (6) is shown in Appendix A. When running this
program, the user is required to enter the time step for
which the equations will be solved explicitly and the initial
pressure, p_o, applied to the hollow sphere. The user may
specify, in the form of data statements in the general
purpose program, the number of subcells in the component
explosive portion of the sphere and, if geometric cell size
expansion is used, the initial cell size and expansion factor
from one cell to the next. The parameters shown in Table 1
are included in the form of data statements in the general
purpose program.

The general purpose program iterates twice through the section which determines $d\lambda(x,t)/dt$, $d\Lambda(r,x,t)/dt$, $T(r,x,t)$, and $dT(r,x,t)/dt$ to converge to the solution. Iterations greater than two do not add significant improvements in convergence. The pressure of the gas is determined by using a JWL (Jones-Wilkens-Lee) Equation of State to model the component explosive gas products. The JWL Equation of State has the following form,

$$p = A \left(1 - \frac{\omega}{R_1 V}\right) \exp(-R_1 V) + B \left(1 - \frac{\omega}{R_2 V}\right) \exp(-R_2 V) + \frac{\omega}{V} E \quad (7)$$

where V, the relative volume, equals v/v_o or ρ_o/ρ. The values for the JWL Equation of State parameters used to model PBX-9404[2] are shown in Table 2.

TABLE 2. JWL EQUATION OF STATE PARAMETERS USED FOR PBX-9404

ρ	1.84	g/cm^3
P_{CJ}	0.37	Mbar
D	0.88	$cm/\mu sec$
E_c	0.102	Mbar-cc/cc
R_1	4.6	
R_2	1.3	
ω	0.38	
A	8.524	Mbar
B	0.1802	Mbar
C	0.0121	Mbar

The relative volume, V, is found by the following method. First one multiplies the volume between r_1 and r_2 by the global burn fraction. This results in a quantity, v_a, which takes into account the relative volume of the high explosive gas products from the component explosive. If v_a is less than 1.E-11, it is set equal to 1.E-11. The current

volume of the hollow sphere, v_{hs}, and the volume of the binder, v_b, are also included in the determination of the relative volume which is given by the following expression,

$$V = \frac{v_{hs} + v_g - v_b}{v_g} \; . \tag{8}$$

Initially, it was desirable to create a model with constants that were solely determined by the physical parameters of the component explosive. However, the use of the JWL Equation of State to model the component explosive gas products introduces another set of unknowns which are usually determined through experiments such as cylinder tests or similitude data. In future applications of the general purpose program for high explosives that do not have JWL Equation of State parameters, it will be assumed that a generic set of JWL Equation of State parameters will be sufficient to model the gas products.

SECTION 3

HOW TO APPLY THE GENERAL PURPOSE
PROGRAM RESULTS TO A HYDRODYNAMIC COMPUTATION

The general purpose program developed in the previous section for the NSWC Burn Model was designed to evaluate one hot spot formation caused by a constant pressure applied at the outer surface of the hollow sphere for the duration of the run. In a one-, two-, or three-dimensional hydrodynamic computation, there are many cells; each cell experiences varying pressure levels throughout the computation. In this section, a scheme for using the results of the general purpose program describing the NSWC Burn Model will be presented.

The first characteristic which needs to be defined is the transition from exponential growth of the reaction due to mechanical deformation to a slower growth of the reaction due to surface burning. This exponential growth of the reaction due to mechanical deformation can be thought of as the ignition phase of a hot spot. The equations and general purpose program presented in Section 2 describe this phenomena. The growth phase in which a transition to surface burning occurs is described with a different, simpler expression for the global burn fraction, such as a surface burn model. This expression is described in more detail in Reference 1.

After evaluating how different system parameters changed as the growth of the reaction slowed, it was observed that the quantity,

11

$$\log_{10}\left(\frac{\lambda(x,t) \, * \, \sum_{j=1}^{n} T(j,x,t) \, * \, dr(j,x,t)}{n \, * \, (r_o - r_i) \, p_o}\right), \qquad (9)$$

changed sign, from a negative to a positive value, when the exponential growth of the reaction due to mechanical deformation slowed for every applied pressure (from 5 to 200 kilobar). The quantity represented by Equation (9) is independent of the time step step selected. In Equation (9), $\lambda(x,t)$ is the global burn fraction, $T(j,x,t)$ is the temperature in the subcell, n is the number of subcells describing the component explosive, $dr(j,x,t)$ is the width of the subcell, r_o and r_i are the current values of the solid portion of the sphere which bound the component explosive region in the model, and p_o is the applied pressure as described in the previous section. When the quantity represented by Equation (9) changed sign in the general purpose program, it was assumed that the global burn fraction reached the transition point, λ_{hs}, at time, t_{hs}, where surface burning becomes the predominant mechanism.

When running a one-, two-, or three-dimensional hydrodynamic computation, another criteria, based on the computation, is used to determine when the transition from ignition to growth should be made. This transition to growth is based upon a minimum burn rate computed from both the ignition model and the growth model. This method of determining transition (presented above) provides enough information to reach and go slightly beyond the criteria of a hydrodynamic computation.

Important consideration also needs to be given when determining the minimum pressure below which a burn fraction will not accumulate in hydrodynamic computations. The general purpose program was run starting at a pressure of

200 kilobar. In each successive run, the pressure was
reduced by 1 kilobar. It was determined that 5 kilobars is
the lowest pressure that produced a smooth transition of λ_{hs}
from one pressure to the next. At 6 kilobars, the value of
λ_{hs} was 4.92e-3; at 5 kilobars, the value of λ_{hs} was 4.15e-3;
and at 4 kilobars, the value of λ_{hs} was 1.22e-8. Therefore,
the lowest acceptable pressure has been selected as
5 kilobars. The data generated at 1 kilobar intervals
includes the global burn fraction at transition, λ_{hs}, and the
associated time at which this occurs, t_{hs} for each pressure.

For pressures of 5 kilobars, pressures of 10 to
100 kilobars at 10-kilobar intervals, and pressures of 100 to
200 kilobars at 20-kilobar intervals, the burn fraction
versus time data is written out for every time step. This
produced a large amount of data which was reduced to be
easily handled by a hydrodynamic computation. The program
which reduces the data, SLIM, is shown in Appendix B. In
addition to data reduction, SLIM also normalizes the data,
determines the length of the line resulting from the
normalized data points, and determines the coordinates of
points which divide the normalized line up into 400 equally
spaced segments based on the original normalized data points.
This allows the hydrodynamic computation to interpolate
between two ranges of pressure to generate a normalized burn
fraction versus time curve for any pressure. It also can be
used to find the new burn fraction.

After much investigation, the following normalization
scheme was found,

$$\log_{10}\left(\left(\frac{X}{X_{h.s.}}\right)^{\frac{P_{c.j.}^Y}{P}}\right) , \tag{10}$$

where Y is 0.15 when X is the time, Y is 0.05 when X is the
burn fraction, P is the pressure in the cell, and P_{cj} is

13

the Chapman-Jouguet pressure of the high explosive. This
normalization scheme seemed to work fairly well in creating
curves from which one could interpolate. The normalized
curves for 5 kilobars, 10 through 100 kilobars at 10-kilobar
intervals, and 100 through 200 kilobars at 20-kilobar
intervals are shown in Figure 2.

In order to use this information in a hydrodynamic
computation, one tabulates the normalized curves in the form
of data statements so that a hydrodynamic computation can
determine the required information. The two programs, MAKEHS
and MAKELAM, used to create the necessary data statements are
shown in Appendix C. A sample subroutine which uses this
information is shown in Appendix D. Currently, this
subroutine is used in the Eulerian hydrocode DYSMAS.

The first time a cell experiences a pressure greater
than a predetermined minimum, such as 5 kilobars for the case
of PBX-9404, the value of the normalized time is determined
using the time step as t, the current pressure in the cell as
p_o, t_{hs} based on current p_o, and the Chapman-Jouguet pressure
of the component explosive. From this, the associated burn
fraction and burn rate can be determined. On subsequent
cycles, the previous burn fraction is used to determine an
old normalized time based on the current value of p_o. Then
the current time step is used to determine a new normalized
time and associated burn fraction and burn rate. Using this
method, it is quite feasible to model the ignition phase with
the NSWC Burn Model in conjunction with a simple surface burn
growth phase model, and to run one-, two-, or three-
dimensional hydrodynamic computations.

SECTION 4

CONCLUSIONS AND FUTURE WORK

The development of a general purpose program to
incorporate the concepts of the NSWC Burn Model into a
tabular data set have been successful. Future work will
involve comparing results from inclusion of the tabulated
data set into various hydrocodes to results shown in
Reference 1. To complete this work, constants in the surface
burn model may need to be modified to match the results shown
in Reference 1.

Future work also will involve (1) verifying the
assumption (discussed in Section 2) that a generic JWL
Equation of State is sufficient because the parameters of the
NSWC Burn Model will compensate as needed, and (2)
investigating whether the criteria for transition from
ignition to growth phase will always be the quantity, shown
in Equation (9), changing sign for different explosive
components and/or similar explosive components with varying
particle sizes, initial porosities, initial temperatures,
and/or initial mass distributions between the component
explosive and binder.

Planned improvements to the general purpose program
include (1) implementing a numerical expression for the bulk
compressibility of the binder to take into account the energy
loss of compressing the binder, and (2) investigating a
method of including shear-induced hot spot temperature
effects.

A - COMPONENT EXPLOSIVE
B - BINDER
C - PORE (VOID)
D - AREAS OF HIGH MECHANICAL
    DEFORMATION (HOT SPOTS)
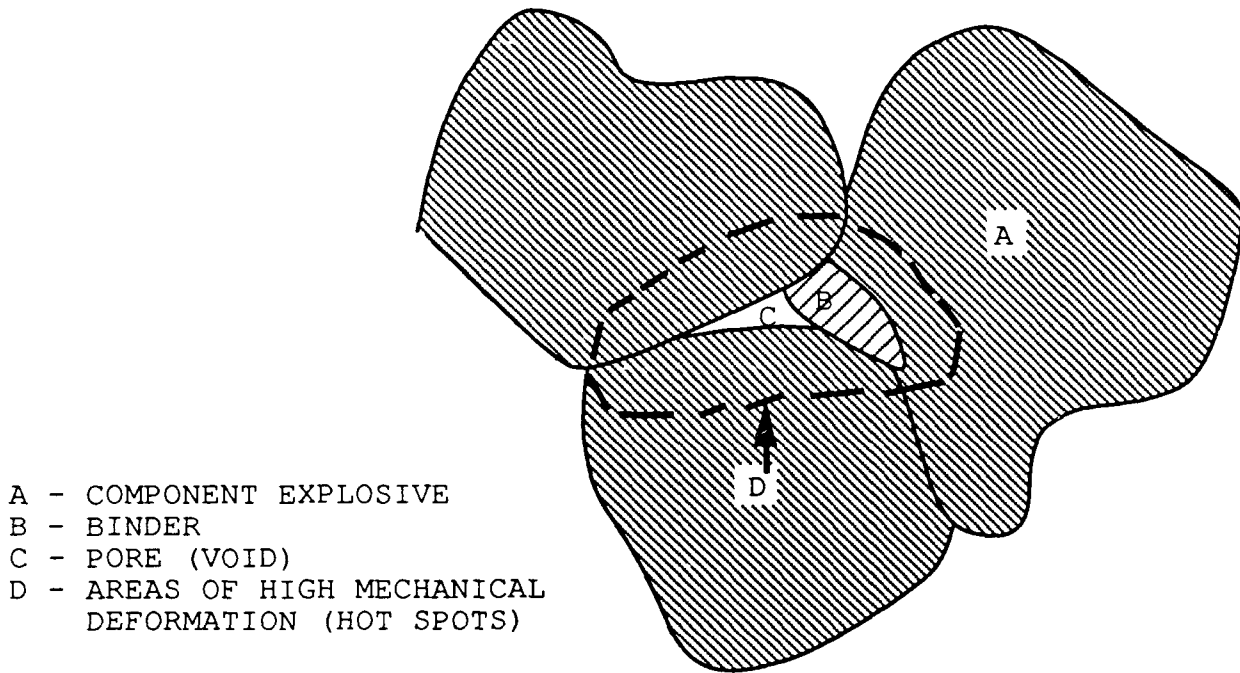
(a) real situation in composite explosive
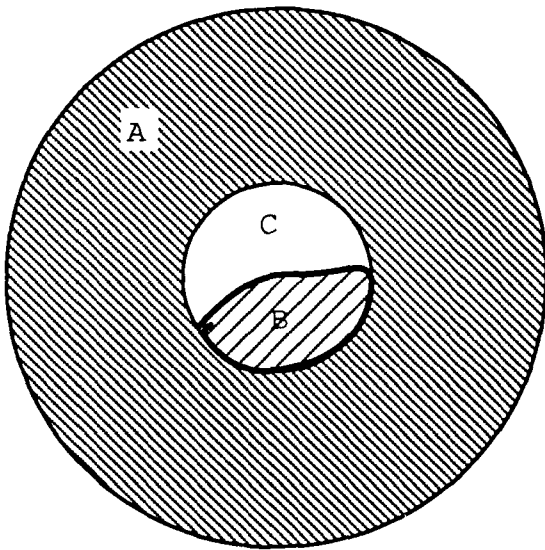
(b) one-dimensional approach
    of the NSWC Burn Model
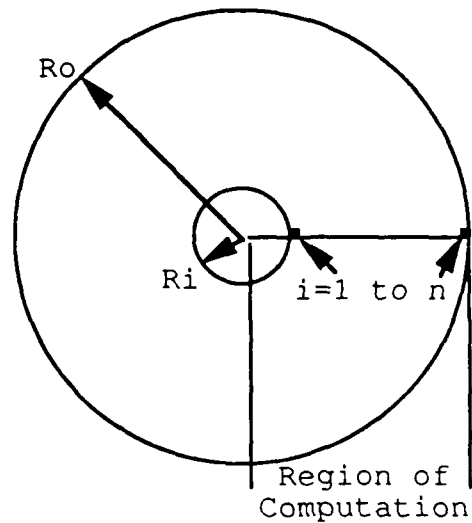
(c) schematic of one-dimensional
    computational region of the
    NSWC Burn Model

FIGURE 1. THE NSWC BURN MODEL

$$\sum LOG_{10}\left[\left(\frac{t}{t_{H.S.}}\sqrt{\left(\frac{P_{CJ}}{P}\right)^{0.1}}\right)\right]$$
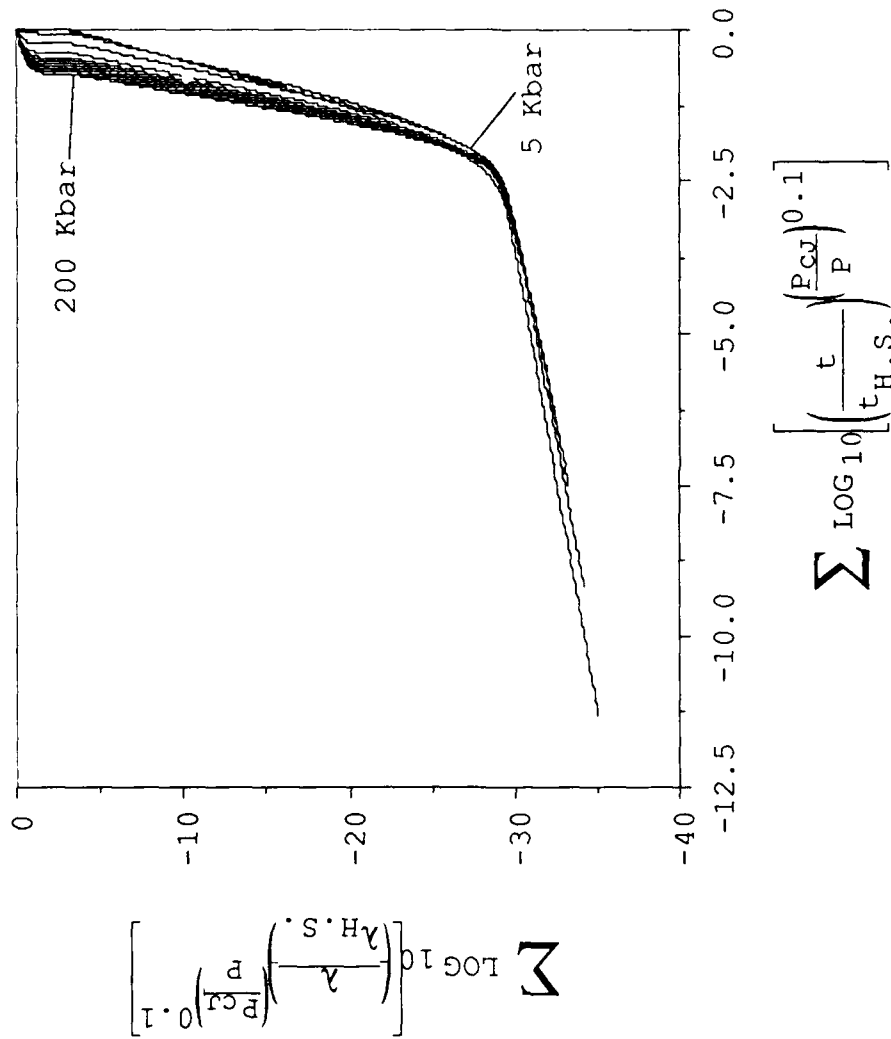
NOTE: LINES CORRESPONDING TO INITIAL PRESSURES OF 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 120, 140, 160, AND 180 KILOBARS BOUNDED BY THE LINES CORRESPONDING TO 5 AND 200 KILOBARS

FIGURE 2. NORMALIZED DATA AT PRESSURES OF 5, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 120, 140, 160, 180, AND 200 KILOBARS FOR PBX-9404

REFERENCES

1. Kim, K., "Development of a Model of Reaction Rates in Shocked Composite Explosives," in Proceedings of the 9th Symposium (International) on Detonation (Preprints), Vol. 3, pp. 946-957, 1989.

2. Dobratz, B. M. and Crawford, P. C., LLNL Explosives Handbook: Properties of Chemical Explosives and Explosive Simulants, UCRL-52997, 31 Jan 1985.

APPENDIX A

GENERAL PURPOSE PROGRAM WHICH
IMPLEMENTS THE CONCEPTS OF THE NSWC BURN MODEL

```
c  3/4/91 version
      program test
      implicit double precision(a-h,o-z)
      dimension temp(450),clam(450),drrl3(450),rm(450),
     1   tburn(450),dslmax(450),tempold(450),rmold(450)
      dimension rbd(451),rbd3(451),dr(450)
      data m/415/,factin/1.01/,cellin/0.000001/
      data z/5.e13/, tstar/2.65e4/, tambi/300./, rho/1.84/
      data cp/1.4e-5/, bulk/8.e-5/, q/5.439e-2/, stark/8.e-14/
      data ri/0.0039/, ro/0.01/, thrd/0.33333333/
      data ajwl/8.524011/, bjwl/0.1801812/, omjwl/0.38/
      data rljwl/4.6/, r2jwl/1.3/, cjwl/0.012066373/
      write(6,*) 'input time step'
      read(5,*) dto
      compold=10000.
      totlen=ro-ri
      gam=0.025
      po=25.e-3
      type *,'Enter initial pressure'
      read(*,*)po
      ms=1+(m-1)/10
      mpl=m+1
      nc=0
      tt=0.
c hot spots will not form if it takes longer than 1 microseconds
      t1=1.
      dt=dto
      ib=1
      istr=1
      rg=ri
      rg3=rg*rg*rg
      rbd(1)=ri
      rbd3(1)=rbd(1)**3
      ro3=ro**3
      ri3=rbd3(1)
      ro3mri3=ro3-ri3
      fako=2.25*gam/(rho*cp*bulk*(1./ri3-1./ro3)**2)
      fakr=3./(ro3mri3)
      fakp=po-3.464102*bulk*dlog(ro/ri)
      fakq=q/cp
      faks=1./cp/rho
      fakv=gam/2./bulk
      prev = cellin/factin
```

```
      sum = ri
      do i=1,m
        ip1=i+1
        prev=factin*prev
        dr(i)=prev
        if(sum+prev.lt.ro)sum=sum+prev
        if(i.eq.m)dr(i)=ro-sum
        temp(i)=tambi
        tempold(i)=tambi
        clam(i)=1.e-30
        tburn(i)=0.
        dslmax(i)=0.
        rbd(ip1)=dr(i)+rbd(i)
        rbd3(ip1)=rbd(ip1)**3
        drrl3(i)=rbd3(ip1)-rbd3(i)
        rm(i)=rbd(ip1)-0.5*dr(i)
        rmold(i)=rm(i)
      end do
   10 continue
c
c     determine volume of gas if not first time through
c
      vo=slam*(rbd3(mp1)-rbd3(1))
      if(vo.lt.1.d-11)vo=1.d-11
      vbold=vb
      vb=0.01*rbd3(mp1)
      spvolg=(rbd3(1)+vo-vb)/vo
      pgold=pg
      pg=ajwl*dexp(-r1jwl*spvolg)+bjwl*dexp(-r2jwl*spvolg)+
     1   cjwl*spvolg**(-omjwl-1)
      if(istr.eq.1)pgbaseline=pg
      if(istr.eq.1)pg=1.01323e-6
      if(istr.ne.1)pg=pg+1.01323e-6-pgbaseline
c
c     determine new radial positions based on velocity
c
      if(istr.ne.1)then
        riold=rbd(1)
        roold=rbd(mp1)
        do i=1,mp1
          v=fakv*(fakp-pg)/(1/ro3-1/ri3)/rbd(i)**2
          rbd(i)=rbd(i)+v*dt
          rbd3(i)=rbd(i)**3
        end do
        ro=rbd(mp1)
        ri=rbd(1)
        totlen=ro-ri
        ro3=rbd3(mp1)
        ri3=rbd3(1)
        ro3mri3=ro3-ri3
        do i=1,m
          dr(i)=rbd(i+1)-rbd(i)
          rm(i)=rbd(i)+dr(i)/2.
        end do
        fakp=po-3.464102*bulk*dlog(ro/ri)
        fako=2.25*gam/(rho*cp*bulk*(1./ri3-1./ro3)**2)
        fakr=3./(ro3mri3)
```

```
         endif
         tt=tt+dt
         dslam=0.0
         tempave=0.0
         dtempave=0.0
         do i=1,m
          if(clam(i).eq.1.d0)goto 55
          r=rm(i)
          r2dr=r*r*dr(i)
          clamold=clam(i)
          dslamold=dslam
          if(i.ne.1.and.i.ne.m)then
             deltempr=(tempold(i+1)-tempold(i))/(rmold(i+1)-rmold(i))
             deltempl=(tempold(i)-tempold(i-1))/(rmold(i)-rmold(i-1))
          elseif(i.eq.1)then
             deltempr=(tempold(i+1)-tempold(i))/(rmold(i+1)-rmold(i))
             deltempl=0.0
          elseif(i.eq.m)then
             deltempr=0.0
             deltempl=(tempold(i)-tempold(i-1))/(rmold(i)-rmold(i-1))
          endif
          subdtempr=deltempr*((rm(i+1)+rm(i))/2.)**2
          subdtempl=deltempl*((rm(i)+rm(i-1))/2.)**2
          subdtemp=faks*stark*(subdtempr-subdtempl)/r2dr
             do 54 jj=1,2
                tdt=tstar/temp(i)
                if(tdt.gt.100.)type *,'tstar/temp(i)',tdt
                if(tdt.gt.100.)tdt=100.
                dclam=(1.-clam(i))*z*dexp(-tstar/temp(i))
                dtemp=fako*(fakp-pg)**2/r**6+fakq*dclam+subdtemp
                clam(i)=dmin1(1.d0,clamold+dclam*dt)
                if(clam(i).eq.1.d0)then
                   dclam=(1.-clamold)/dt
                   dtemp=fako*(fakp-pg)**2/r**6+fakq*dclam+subdtempr+subtempl
                endif
                temp(i)=tempold(i)+dtemp*dt
                dslam=dslamold+fakr*dclam*r2dr
                if(clam(i).eq.1.d0)goto 55
54          continue
55          continue
         tempave=tempave+temp(i)*dr(i)
         dtempave=dtempave+dtemp*dr(i)
         end do
         istr=0
         slam=slam+dslam*dt
         do 56 i=1,m
            tempold(i)=temp(i)
            rmold(i)=rm(i)
56       continue
         rg3=rbd3(ib)+clam(ib)*drrl3(ib)
         rg=rg3**thrd
         if(clam(ib).ge.0.999999d0) then
            tburn(ib)=tt
            dslmax(ib)=dslam
            ib=ib+1
         end if
         nc=mod(nc+1,10)
```

```
      mss=5
      mdum=25
      comp=tempave/m/totlen*slam/po
      if(comp.eq.0.)comp=compold-1.
      comp=dlog10(comp)
c   hot spot lambda is reached according to criteria on next line
      if(comp.gt.0.0)goto 35
      compold=comp
      slamold=slam
       if(po.eq.0.01.or.po.eq.0.02.or.po.eq.0.03.or.po.eq.0.04
     1 .or.po.eq.0.05.or.po.eq.0.06.or.po.eq.0.07
     2 .or.po.eq.0.08.or.po.eq.0.09.or.po.eq.0.10
     3 .or.po.eq.0.11.or.po.eq.0.12.or.po.eq.0.13
     4 .or.po.eq.0.14.or.po.eq.0.15.or.po.eq.0.005)
     1  write(8,'(1x,1p,5e13.5)')slam,dslam,comp,tt,pg
      if(tt.lt.t1.and.clam(m).lt.1.) goto 10
c   the next line is designed to crash the batch job when t hot spot
greater than
c   1 microsecond.
      if(tt.ge.t1)acrash=1./0.
35    continue
      open(22,file='hs.info',status='unknown')
      do 838 ids=1,10000
        read(22,*,end=839,err=839)a,b,c
838   continue
839   continue
      write(22,'(7(1x,e15.7))')
     1      po,tt-dt,slamold,pgold,riold,roold,vbold
      close(22)
      type *,'t hot spot = ',tt-dt,' slam = ',slamold
      stop
      end
```

APPENDIX B

PROGRAM TO REDUCE AND NORMALIZE
OUTPUT FROM THE GENERAL PURPOSE PROGRAM

```
c   compile with for/g_float
        implicit double precision(a-h,o-z)
        dimension asave(1000),dsave(1000)
        character*14 afile
        character*10 bfile
        aold=1000000.
        type *,'enter file name to read'
        read(*,100)afile
100     format(a14)
        bfile(1:1)=afile(1:1)
        bfile(2:10)=afile(6:14)
        open(82,file=afile,status='old')
        open(84,file=bfile,status='new')
        inum=0
        do 120 i=1,1000000
        read(82,*,end=999)a,b,c,d,e
        anew=dlog10(a)
        if(dabs(anew-aold).gt..1.or.
     1    (a.gt.0.0001.and.dabs(anew-aold).gt..01))then
          inum=inum+1
          if(inum.gt.1000)stop
          asave(inum)=a
          dsave(inum)=d
          ipr=1
          aold=anew
        else
          ipr=0
        endif
120     continue
999     continue
        if(ipr.eq.0)then
          inum=inum+1
          if(inum.gt.1000)stop
          asave(inum)=a
          dsave(inum)=d
        endif
        type *,'what is pressure,pcj'
        read(*,*)pres,pcj
        quanx=(pcj/pres)**.15
        quany=(pcj/pres)**.05
c   convert to scaled quantities
        do 200 iic=1,inum
          asave(iic)=dlog10((asave(iic)/asave(inum))**quany)
```

```
            dsave(iic)=dlog10((dsave(iic)/dsave(inum))**quanx)
200     continue
c   find total length
        inumm1=inum-1
        sumlen=0.
        do 210 iic=1,inumm1
          sumlen=sumlen+dsqrt((asave(iic+1)-asave(iic))**2+
     1                        (dsave(iic+1)-dsave(iic))**2)
210     continue
        icar=2
        icarm1=icar-1
        dellen=sumlen/400.
        curlen=0.0
        ruflen=0.0
        iflr=1
        write(84,431)dsave(1),asave(1)
        do 220 i=1,399
          curlen=curlen+dellen
230       if(iflr.eq.1)ruflen=ruflen+
     1        dsqrt((asave(icar)-asave(icarm1))**2+
     2              (dsave(icar)-dsave(icarm1))**2)
          if(curlen.gt.ruflen)then
            icarm1=icar
            icar=icar+1
            iflr=1
            goto 230
          endif
          iflr=0
          dbtp=dsqrt((asave(icar)-asave(icarm1))**2+
     2              (dsave(icar)-dsave(icarm1))**2)
          ari=dbtp-(ruflen-curlen)
          aval=ari*(asave(icar)-asave(icarm1))/dbtp+asave(icarm1)
          dval=ari*(dsave(icar)-dsave(icarm1))/dbtp+dsave(icarm1)
          write(84,431)dval,aval
220     continue
431     format(1x,f11.7,4x,f11.7)
        write(84,431)dsave(inum),asave(inum)
        stop
        end
```

APPENDIX C

PROGRAMS TO CREATE FORTRAN DATA
STATEMENTS FROM THE TABULATED DATA SET
FOR INCLUSION INTO A SUBROUTINE CALLED BY
ANY ONE-, TWO-, OR THREE-DIMENSIONAL HYDROCODE

```fortran
      program makehs
      parameter iam=500
      dimension po(iam),ths(iam),alamhs(iam),pg(iam),ri(iam),
     1          ro(iam),vb(iam)
      open(81,file='hs.info',status='old')
      icou=0
      do 100 i=1,1000
      read(81,*,end=999)po(i),ths(i),alamhs(i),pg(i),ri(i),
     1                  ro(i),vb(i)
      icou=icou+1
100   continue
999   continue
      open(82,file='hsdata.for',status='new')
      write(82,110)icou,icou,icou,icou,icou
110   format(6x,'dimension po(',i3,'),ri(',i3,'),ro(',i3,
     1        '),',/,5x,'1  alamhs(',i3,'),ths(',i3,')')
      write(82,115)icou
115   format(6x,'data nrs/',i3,'/')
      do 200 i=1,icou
      write(82,120)i,ri(i),i,ro(i),i,po(i)
120   format(6x,'data ri(',i3,')/',e13.7,
     1   '/,ro(',i3,')/',e13.7,'/,po(',i3,')/',f5.3,'/')
200   continue
      do 210 i=1,icou
      write(82,130)i,alamhs(i),i,ths(i)
130   format(6x,'data alamhs(',i3,')/',e13.7,'/, ths(',i3,')/',
     1         e13.7,'/')
210   continue
      stop
      end
```

```
       program makelam
       implicit double precision (a-h,o-z)
       character*10 filename
       parameter iam=1000
       dimension adum(iam),tdum(iam)
       numfil=16
       pcj=.370
       filename(1:2)='f8'
       filename(7:10)='.dat'
       open(82,file='lamfile.for',status='new')
       do 90 iiu=1,numfil
       if(iiu.eq.12)isep=iiu
       if(iiu.gt.12)isep=isep+2
       if(iiu.eq.16)filename(3:6)='p200'
       if(iiu.eq.15)filename(3:6)='p180'
       if(iiu.eq.14)filename(3:6)='p160'
       if(iiu.eq.13)filename(3:6)='p140'
       if(iiu.eq.12)filename(3:6)='p120'
       if(iiu.eq.11)filename(3:6)='p100'
       if(iiu.eq.10)filename(3:6)='p090'
       if(iiu.eq.9)filename(3:6)='p080'
       if(iiu.eq.8)filename(3:6)='p070'
       if(iiu.eq.7)filename(3:6)='p060'
       if(iiu.eq.6)filename(3:6)='p050'
       if(iiu.eq.5)filename(3:6)='p040'
       if(iiu.eq.4)filename(3:6)='p030'
       if(iiu.eq.3)filename(3:6)='p020'
       if(iiu.eq.2)filename(3:6)='p010'
       if(iiu.eq.1)filename(3:6)='p005'
       open(81,file=filename,status='old')
       icou=0
       do 100 i=1,1000
       read(81,*,end=999)adum(i),tdum(i)
100    icou=icou+1
999    if(iiu.lt.11)write(82,110)iiu-1,icou,iiu-1,icou
       if(iiu.eq.11)write(82,111)iiu-1,icou,iiu-1,icou
       if(iiu.gt.11)write(82,111)isep,icou,isep,icou
110    format(6x,'dimension al',i1,'0(',i3,'),t',i1,'0(',i3,')')
111    format(6x,'dimension al',i2,'0(',i3,'),t',i2,'0(',i3,')')
       if(iiu.lt.11)write(82,115)iiu-1,icou
       if(iiu.eq.11)write(82,116)iiu-1,icou
       if(iiu.gt.11)write(82,116)isep,icou
115    format(6x,'data n',i1,'0/',i3,'/')
116    format(6x,'data n',i2,'0/',i3,'/')
       do 200 i=1,icou
       if(iiu.lt.11)write(82,120)iiu-1,i,adum(i),iiu-1,i,tdum(i)
       if(iiu.eq.11)write(82,121)iiu-1,i,adum(i),iiu-1,i,tdum(i)
       if(iiu.gt.11)write(82,121)isep,i,adum(i),isep,i,tdum(i)
120    format(6x,'data al',i1,'0(',i3,')/',e13.7,
      1    '/,t',i1,'0(',i3,')/',e13.7,'/')
121    format(6x,'data al',i2,'0(',i3,')/',e13.7,
      1    '/,t',i2,'0(',i3,')/',e13.7,'/')
200    continue
       close(81)
90     continue
       stop
       end
```

APPENDIX D

SAMPLE OF SUBROUTINE TO BE
INCLUDED INTO A ONE-, TWO-, OR THREE-DIMENSIONAL
HYDRODYNAMIC COMPUTATION TO DETERMINE A NEW BURN FRACTION

```
CDECK USRDFB
       FUNCTION USRDFB(FBRN,TIM,DT,PRS,SMP,NP,I,J,K,IMAX,JMAX,KMAX,
     1                 SDETIN,PDETIN,PBREF,BUSR,CUSR,DUSR)
C
C  ****************************************************************
C
C  *** THIS IS A SAMPLE USER-INPUT FUNCTION FOR THE EVALUATION
C      OF EXPLOSIVE DECOMPOSITION
C
C  *** THIS MODULE IS ACCESSED WHENEVER BURN MODEL KBURN=-9 IS CHOSEN
C      FROM INPUT
C
C  *** DEFINITION OF INPUT PARAMETERS -
C         USRDFB   OUT   CURRENT VALUE OF BURN RATE
C         FBRN     INP   'OLD' VALUE OF BURN FRACTION
C                  OUT   'NEW' VALUE OF BURN FRACTION
C         TIM      INP   CURRENT TIME
C         DT       INP   CURRENT TIME STEP
C         PRS      INP   CURRENT LOCAL PRESSURE
C         SMP      INP   CURRENT LOCAL TEMPERATURE
C         NP       INP   CURRENT EXTERNAL MATERIAL NUMBER
C         I        INP   CELL INDEX  (I-DIRECTION)
C         J        INP   CELL INDEX  (J-DIRECTION)
C         K        INP   CELL INDEX  (K-DIRECTION)
C         IMAX     INP   MAXIMUM NUMBER CF CELLS  (I-DIRECTION)
C         JMAX     INP   MAXIMUM NUMBER OF CELLS  (J-DIRECTION)
C         KMAX     INP   MAXIMUM NUMBER OF CELLS  (K-DIRECTION)
C         SDETIN   INP   USER DEFINED - SUGGESTED: INITIATION TIME
C         PDETIN   INP   USER DEFINED - SUGGESTED: INITIATION PRESSURE
C         PBREF    INP   USER DEFINED - SUGGESTED: REFERENCE PRESSURE
C         BUSR     INP   USER DEFINED - SUGGESTED: NONE
C         CUSR     INP   USER DEFINED - SUGGESTED: NONE
C         DUSR     INP   USER DEFINED - SUGGESTED: NONE
C
C
C
C  *** THIS ROUTINE SHOULD RETURN -
C         EITHER  A BURN RATE BY MEANS OF THE FUNCTION NAME
C         OR      AN UPDATED VALUE OF THE BURN FRACTION VIA FBRN
C      THE APPROPRIATE DEFINITION IS DONE DEPENDING ON -
C         IF ON OUTPUT FBRN IS FOUND TO HAVE CHANGED ITS VALUE,
C         THEN THE UPDATED VALUE OF FBRN IS USED
```

```
C           (IN THIS CASE IT IS ADVISABLE TO DEFINE USRDFB=0.)
C           OTHERWISE A BURN RATE DEFINITION IS ASSUMED
C
C     ***********************************************************************
      IF(PRS.LT.5.E9)RETURN
      SFB=FBRN
      ZZFD=0.0
      IF(FBRN.GE.1.E-5)ZZFD=FBRN
      CALL BEEN(ZZFD,TIM,DT,PRS,SMP,NP,I,J,K,IMAX,JMAX,KMAX,
     1            SDETIN,PDETIN,PBREF,BUSR,CUSR,DUSR)
      IF(ZZFD.GE.1.E-5)THEN
        FBRN=ZZFD
        URSDFB=0.0
      ELSE
        FBRN=SFB
        URSDFB=1.E-20
      ENDIF
      RETURN
      END
      SUBROUTINE BEFN(FBRN,TIM,DT,PRS,SMP,NP,I,J,K,IMAX,JMAX,KMAX,
     1            SDETIN,PDETIN,PBREF,BUSR,CUSR,DUSR)
      DATA CV/2.09E-5/, AJWL/8.524011/, BJWL/0.1801812/, OMJWL/0.38/
      DATA R1JWL/4.6/, R2JWL/1.3/, CJWL/0.012066373/
      DATA TDT/.6666667/,TLIM/1.E-3/
      INCLUDE 'LAMFILE.FOR'
      INCLUDE 'HSDATA.FOR'
      common/icjckc/ IC(1000),JC(1000),KC(1000),FCC(1000)
C PUT DT INTO MICROSECOND UNITS AND SAVE ORIGINAL VALUE OF DT
      SAVEDT=DT
      DT=DT*1.E6
C IN GENERAL, PCJ SHOULD EQUAL PBREF, THE CJ PRESSURE IN THE INPUT
C IE PCJ = PBREF/1.E12 TO GET TO MEGABARS.
      PCJ=PBREF/1.E12
C PRES IS IN UNITS OF MEGABARS, PRS IN UNITS OF MICROBARS
      PRES=PRS/1.E12
C USE 5 KBAR AS MINIMUM PRESSURE FOR REACTION TO OCCUR BASED ON GENERAL
C PURPOSE PROGRAM RESULTS
C MINIMUM PRESSURE SHOULD CORRESPOND TO VALUE IN PO(NRS) FROM HS.INFO
C THE POINT BELOW WHICH THE GENERAL PURPOSE PROGRAM DOES NOT GO
      IF(PRES.LT.0.00501)THEN
        DT=SAVEDT
        RETURN
      ENDIF
      IGOTOSUR=0
      IF(PRES.GT.0.200)IGOTOSUR=1
C DETERMINE ALAMHS BY LOOKING UP TABLE OF DATA
C IF FBRN GE AHS USE SURFACE BURN, IF FBRN LT AHS USE NSWC MODEL
      IREAD=0
      IPICK=0
      IF(BUSR.LT.1.E10.AND.FBRN.LT.TLIM.AND.FBRN.GE.0.0)THEN
        IBUSR=IFIX(BUSR+0.00001)
        DO 95 KDC=1,IBUSR
        IF(IC(KDC).EQ.I.AND.JC(KDC).EQ.J.AND.KC(KDC).EQ.K)THEN
          IPICK=1
          FBRN=FBRN+FCC(KDC)
        ENDIF
95      CONTINUE
```

```
      IREAD=1
      ENDIF
      SAVEFBRN=FBRN
      IF(IGOTOSUR.EQ.1)GOTO 338
      NRSM1=NRS-1
      DO 400 ICP=1,NRSM1
        IF(PRES.GE.PO(ICP+1).AND.PRES.LT.PO(ICP))THEN
          ICP1=ICP+1
          AHS=(ALAMHS(ICP1)-ALAMHS(ICP))*(PRES-PO(ICP))/
     1         (PO(ICP1)-PO(ICP))+ALAMHS(ICP)
          TAHS=(THS(ICP1)-THS(ICP))*(PRES-PO(ICP))/
     1         (PO(ICP1)-PO(ICP))+THS(ICP)
          GOTO 401
        ENDIF
400   CONTINUE
401   CONTINUE
      IF(FBRN.LT.AHS.AND.FBRN.GE.0.0)THEN
C     USE NSWC MODEL RELATIONS TO FIND FBRN
      IF(PRES.GE.0.18)THEN
      CALL AFORP(AL180,T180,AL200,T200,N180,N200,FBRN,
     1    DT,PRES,PO(21),PO(1),PCJ,THS(21),THS(1),AHS,TAHS)
      ELSEIF(PRES.LT.0.18.AND.PRES.GE.0.16)THEN
      CALL AFORP(AL160,T160,AL180,T180,N160,N180,FBRN,
     1    DT,PRES,PO(41),PO(21),PCJ,THS(41),THS(21),AHS,TAHS)
      ELSEIF(PRES.LT.0.16.AND.PRES.GE.0.14)THEN
      CALL AFORP(AL140,T140,AL160,T160,N140,N160,FBRN,
     1    DT,PRES,PO(61),PO(41),PCJ,THS(61),THS(41),AHS,TAHS)
      ELSEIF(PRES.LT.0.14.AND.PRES.GE.0.12)THEN
      CALL AFORP(AL120,T120,AL140,T140,N120,N140,FBRN,
     1    DT,PRES,PO(81),PO(61),PCJ,THS(81),THS(61),AHS,TAHS)
      ELSEIF(PRES.LT.0.12.AND.PRES.GE.0.1)THEN
      CALL AFORP(AL100,T100,AL120,T120,N100,N120,FBRN,
     1    DT,PRES,PO(101),PO(81),PCJ,THS(101),THS(81),AHS,TAHS)
      ELSEIF(PRES.LT.0.1.AND.PRES.GE.0.09)THEN
      CALL AFORP(AL90,T90,AL100,T100,N90,N100,FBRN,
     1    DT,PRES,PO(111),PO(101),PCJ,THS(111),THS(101),AHS,TAHS)
      ELSEIF(PRES.LT.0.09.AND.PRES.GE.0.08)THEN
      CALL AFORP(AL80,T80,AL90,T90,N80,N90,FBRN,
     1    DT,PRES,PO(121),PO(111),PCJ,THS(121),THS(111),AHS,TAHS)
      ELSEIF(PRES.LT.0.08.AND.PRES.GE.0.07)THEN
      CALL AFORP(AL70,T70,AL80,T80,N70,N80,FBRN,
     1    DT,PRES,PO(131),PO(121),PCJ,THS(131),THS(121),AHS,TAHS)
      ELSEIF(PRES.LT.0.07.AND.PRES.GE.0.06)THEN
      CALL AFORP(AL60,T60,AL70,T70,N60,N70,FBRN,
     1    DT,PRES,PO(141),PO(131),PCJ,THS(141),THS(131),AHS,TAHS)
      ELSEIF(PRES.LT.0.06.AND.PRES.GE.0.05)THEN
      CALL AFORP(AL50,T50,AL60,T60,N50,N60,FBRN,
     1    DT,PRES,PO(151),PO(141),PCJ,THS(151),THS(141),AHS,TAHS)
      ELSEIF(PRES.LT.0.05.AND.PRES.GE.0.04)THEN
      CALL AFORP(AL40,T40,AL50,T50,N80,N90,FBRN,
     1    DT,PRES,PO(161),PO(151),PCJ,THS(161),THS(151),AHS,TAHS)
      ELSEIF(PRES.LT.0.04.AND.PRES.GE.0.03)THEN
      CALL AFORP(AL30,T30,AL40,T40,N30,N40,FBRN,
     1    DT,PRES,PO(171),PO(161),PCJ,THS(171),THS(161),AHS,TAHS)
      ELSEIF(PRES.LT.0.03.AND.PRES.GE.0.02)THEN
      CALL AFORP(AL20,T20,AL30,T30,N20,N30,FBRN,
     1    DT,PRES,PO(181),PO(171),PCJ,THS(181),THS(171),AHS,TAHS)
```

```
      ELSEIF(PRES.LT.0.02.AND.PRES.GE.0.01)THEN
      CALL AFORP(AL10,T10,AL20,T20,N10,N20,FBRN,
     1   DT,PRES,PO(191),PO(181),PCJ,THS(191),THS(181),AHS,TAHS)
      ELSEIF(PRES.LT.0.01)THEN
      CALL AFORP(AL00,T00,AL10,T10,N00,N10,FBRN,
     1   DT,PRES,PO(NRS),PO(191),PCJ,THS(NRS),THS(191),AHS,TAHS)
      ENDIF
      ENDIF
      AFORPFBRN=FBRN
338   CONTINUE
C     IF(FBRN.GE.AHS.AND.FBRN.LT.1.0)THEN
      IF(FBRN.LT.1.0)THEN
      FBRN=SAVEFBRN
C   USE SURFACE BURN RELATIONS TO FIND FBRN
      CALL SURKIM(PRES,FBRN,DT,AHS)
      ENDIF
      SURKIMFBRN=FBRN
      BRATE1=(AFORPFBRN-SAVEFBRN)/DT
      BRATE2=(SURKIMFBRN-SAVEFBRN)/DT
C  FBRN IS CURRENTLY FROM SURKIM, CHANGE IF BRATE1 FBRN FROM AFORP IS
LESS
      IF(BRATE1.LE.BRATE2)FBRN=AFORPFBRN
      DT=SAVEDT
      IF(FBRN.GE.TLIM.AND.IREAD.EQ.0)GOTO 774
      IF(FBRN.EQ.0.0.AND.IREAD.EQ.0)GOTO 774
      IF(FBRN.LT.TLIM.AND.IREAD.EQ.0.AND.FBRN.GT.0.0)THEN
        IF(BUSR.LT.1.E10)BUSR=BUSR+1.
        IF(BUSR.GT.1.E10)BUSR=1.
        IBUSR=IFIX(BUSR+0.00001)
        IC(IBUSR)=I
        JC(IBUSR)=J
        KC(IBUSR)=K
        FCC(IBUSR)=FBRN
      ELSEIF(FBRN.GE.TLIM.AND.IREAD.EQ.1.AND.IPICK.EQ.1)THEN
        IKF=0
        DO 96 KDC=1,IBUSR
           IF(IKF.EQ.1)IC(KDC-1)=IC(KDC)
           IF(IKF.EQ.1)JC(KDC-1)=JC(KDC)
           IF(IKF.EQ.1)KC(KDC-1)=KC(KDC)
           IF(IKF.EQ.1)FCC(KDC-1)=FCC(KDC)
           IF(IC(KDC).EQ.I.AND.JC(KDC).EQ.J.AND.KC(KDC).EQ.K)IKF=1
96      CONTINUE
        IBUSR=IBUSR-1
        BUSR=BUSR-1.
      ELSEIF(FBRN.LT.TLIM.AND.IREAD.EQ.1.AND.IPICK.EQ.1)THEN
        DO 97 KDC=1,IBUSR
        IF(IC(KDC).EQ.I.AND.JC(KDC).EQ.J.AND.KC(KDC).EQ.K)
     1      FCC(KDC)=FBRN
97      CONTINUE
      ELSEIF(FBRN.LT.TLIM.AND.IREAD.EQ.1.AND.IPICK.EQ.0)THEN
        BUSR=BUSR+1
        IBUSR=IFIX(BUSR+0.00001)
        IC(IBUSR)=I
        JC(IBUSR)=J
        KC(IBUSR)=K
        FCC(IBUSR)=FBRN
      ENDIF
```

```
774       CONTINUE
          RETURN
          END
          SUBROUTINE AFORP(A1,T1,A2,T2,N1,N2,FBRN,DT,PRES,PRES1,PRES2,PCJ,
      1                     THS1,THS2,AHS,TAHS)
C FBRN:      INPUT -OLD BURN FRACTION
C           OUTPUT -NEW BURN FRACTION
          DIMENSION A1(N1),T1(N1),A2(N2),T2(N2),AAVE(1000),TAVE(1000)
          INCLUDE 'HSDATA.FOR'
          IF(N1.NE.N2)STOP 123
          QUANX=(PCJ/PRES)**.15
          QUANY=(PCJ/PRES)**.05
          N1M1=N1-1
          DO 100 IR=1,N1
            AAVE(IR)=(A2(IR)-A1(IR))*(PRES-PRES1)/(PRES2-PRES1)+A1(IR)
            TAVE(IR)=(T2(IR)-T1(IR))*(PRES-PRES1)/(PRES2-PRES1)+T1(IR)
100       CONTINUE
          IF(FBRN.EQ.0.0)THEN
            TBN=DT
          ELSE
            FBNSCL=ALOG10((FBRN/AHS)**QUANY)
C   NEED TO FIND OLD TIME
            DO 110 I=1,N1M1
            IF(AAVE(I).LT.FBNSCL.AND.AAVE(I+1).GE.FBNSCL)THEN
              TBOSCL=(FBNSCL-AAVE(I))*(TAVE(I+1)-TAVE(I))/
      1               (AAVE(I+1)-AAVE(I))+TAVE(I)
              GOTO 111
            ENDIF
110         CONTINUE
111         CONTINUE
            TBO=TAHS*((10.**TBOSCL)**(1./QUANX))
            TBN=TBO+DT
          ENDIF
          IF(TBN.GT.TAHS)THEN
            FBRN=AHS
            USEDT=TAHS-TBO
            SDT=DT-USEDT
            IF(SDT.GT.0.0)DT=SDT
            RETURN
          ELSE
            TBNSCL=ALOG10((TBN/TAHS)**QUANX)
            DO 120 I=1,N1M1
            IF(TAVE(I).LT.TBNSCL.AND.TAVE(I+1).GE.TBNSCL)THEN
              FBNSCL=(TBNSCL-TAVE(I))*(AAVE(I+1)-AAVE(I))/
      1               (TAVE(I+1)-TAVE(I))+AAVE(I)
              GOTO 121
            ENDIF
120         CONTINUE
121         CONTINUE
          ENDIF
          FBRN=AHS*((10.**FBNSCL)**(1./QUANY))
          RETURN
          END
          SUBROUTINE SURKIM(PRES,FBRN,DT,AHS)
          DATA CV/2.09E-5/, AJWL/8.524011/, BJWL/0.1801812/, OMJWL/0.38/
          DATA R1JWL/4.6/, R2JWL/1.3/, CJWL/0.012066373/
          DATA TDT/.6666667/,ALP/1.2/,BET/-0.4/,DEL/32500./,EP/2.5/
```

```
      DATA ROORG/.01/,RIORG/.0039/,ANVAL/0.9/
      SBR=3.*.00627/ROORG
      DSLAM=SBR*(FBRN**TDT*ALP*PRES**(BET+ANVAL)+
     1         (1.-FBRN)**TDT*DEL*PRES**(EP+ANVAL))
      FBRN=FBRN+DT*DSLAM
      FBRN=AMIN1(FBRN,1.)
      RETURN
      END
```

DISTRIBUTION

	Copies		Copies
OUSDRE/R&AT-MST Attn: R. Siewart The Pentagon Washington, DC 20301	1	Institute for Defense Analyses Attn: Technical Library 1801 N. Beauregard St. Alexandria, VA 22311	1
OUSDRE/TWP-NW&M Attn: D. Anderson The Pentagon Washington, DC 20301	1	Director Defense Nuclear Agency Attn: Technical Library Washington, DC 20305	1

OUSDRE/TWP-CM
Attn: G. Kopcsak
The Pentagon
Washington, DC 20301
	1	Chairman
		Department of Defense
		  Explosives Safety Board
		Attn: 6-A-145
		    DDESB-KT
			1
			1

Defense Science Board Attn: C. Fowler The Pentagon Washington, DC 20301	1	Hoffman Building 1 2461 Eisenhower Avenue Alexandria, VA 22331	

USD(A)/DDRE (R/AT/ET)
Staff Specialist for Weapons
  Technology
Attn: R. Menz
The Pentagon
Washington, DC 20301
	1	OASN/RE&S
		  - Surface Warfare
		  - Air Warfare
		  - Subs/ASW
		Navy Department
		Washington, DC 20301
			1
			1
			1

Defense Technical Information
  Center
Cameron Station
Alexandria, VA 22304-6145
	2	Office of the Chief of
		  Naval Research
		Attn: ONR 1132P (R. Miller)
		    ONT 20T (L. V. Schmidt)
		    ONT 21 (E. Zimet)
		    ONT 213 (D. Siegel)
		    ONT 23 (A. Faulstich)
		    ONT 232 (D. Houser)
			1
			1
			1
			1
			1
			1

Director Defense Research and Engineering Attn: Library Washington, DC 20305	1	800 N. Quincy Street, BCT 1 Arlington, VA 22217-5000	
Director Defense Advanced Research Projects Agency Attn: Library 1400 Wilson Blvd. Arlington, VA 22209	1	Commander Center for Naval Analyses Attn: Technical Library 2000 N. Beauregard Street Alexandria, VA 22311	1

	Copies			Copies
Commanding Officer Naval Research Laboratory Attn: Technical Library Washington, DC 20375	1	Commander Naval Weapons Center Attn: Code 38 (R. Derr) Code 389 (T. Boggs) Code 32 (E. Lundstrom Code 326B (L. Josephson) Technical Library China Lake, CA 93555	1 1 1 1 1	
Commander Naval Air Systems Command Attn: Technical Library Washington, DC 20361	1			
Commander Naval Sea Systems Command Attn: SEA-66I (R. Bowen) Washington, DC 20362-5105	1	Commanding Officer Naval Amphibious Base, Coronado Attn: RDT Officer SEAL Team Underwater Demolition Team San Diego, CA 92155	1 1 1	
Commander Space and Naval Warfare Systems Command Attn: SPAWAR-05 Washington, DC 20363-5100	1	Commanding Officer Naval Amphibious Base, Little Creek Attn: RDT Officer Norfolk, VA 23511	1	
Commanding Officer Naval Coastal Systems Center Attn: Technical Library Panama City, FL 32407-5000	1	Commanding General Marine Corps Development and Education Command Attn: Library Marine Corps Landing Force Development Center Quantico, VA 22134	1	
Commander David Taylor Research Center Attn: Technical Library S. Wang Bethesda, MD 20084	1 1			
Commander David Taylor Research Center Attn: Code 177 Technical Library Portsmouth, VA 23709	1 1	Commanding Officer Naval Explosive Ordnance Disposal Technology Center Attn: Technical Library Indian Head, MD 20640	1	
Commander Naval Ocean Systems Center Attn: Technical Library San Diego, CA 92152	1	Commanding Officer Naval Intelligence Support Center 4301 Suitland Road Washington, DC 20390-5140	1	
Commander Naval Underwater Systems Center Attn: Technical Library Code 363 (R. Nadolink) Newport, RI 02840	1 1	Commander Pacific Missile Test Center Attn: Code 2145 Point Mugu, CA 93042	1	

Copies

Copies

Commanding Officer
Naval Ordnance Station
Attn: Code 2730D   1
       Technical Library   1
       J. Chang   1
       P. Dendor   1
       L. Newman   1
Indian Head, MD 20640-5000

Superintendent
Naval Postgraduate School
Attn: Library   1
Monterey, CA 93940

Commanding Officer
SEAL Team 2
FPO New York 09501-4633   1

Commanding Officer
Naval Ship Weapons Systems
   Engineering Station
Port Hueneme, CA 93043   1

Commanding Officer
Naval Undersea Warfare
   Engineering Station
Keyport, WA 98345   1

President
Naval War College
Attn: Technical Library   1
Newport, RI 02841

Commander
Naval Weapons Evaluation Facility
Kirtland Air Force Base
Albuquerque, NM 87117   1

Commanding Officer
Naval Weapons Station
Attn: Code 321 (M. Bucher)   1
Concord, CA 94520-5000

Commanding Officer
Naval Weapons Station
Attn: Code 470A   1
       Library   1
Yorktown, VA 23691-5000

Commanding Officer
Naval Weapons Support Center
Attn: Code 3031 (E. Neal)   1
       Code 50D (A. Norris)   1
       Code 505 (J. E. Short)   1
       Code 90 (A. E. Whitner)   1
Crane, IN 47522

Army Armament Munitions and
   Chemical Command
Attn: DRSAR-IRC   1
       DRSAR-LEM (R. Freeman)   1
       DRSAR-SF (R. Young)   1
Rock Island, IL 61299-6000

U.S. Army Armament Research,
   Development and Engineering Center
Attn: Technical Library   1
Dover, NJ 07806-5000

U. S. Army Ballistic Research
   Laboratory
Attn: DRDAR-BLP   1
       DRDAR-BLT (R. Frey)   1
       SLCRBR-IB-I (P. Kaste)   1
       Technical Library   1
Aberdeen Proving Ground, MD
21005

Commander
USA Chemical Research, Development
   and Engineering Center
Attn: SMCCR-DDP   1
Aberdeen Proving Ground, MD
21010-5423

Army Environmental Hygience Agency
Attn: HSHB-EA-A   1
Aberdeen Proving Ground, MD
21005

Commanding Officer
Harry Diamond Laboratories
Attn: Library   1
2800 Powder Mill Road
Adelphi, MD 20783

DISTRIBUTION (Cont.)

Copies | | Copies

Army Medical Bioengineering
  Research and Development
  Laboratory
Attn: J. Barkeley   1
Fort Dietrick, MD 21701

Commander
U.S. Army Research Office
Attn: G.R. Husk   1
P.O. Box 12211
Research Triangle Park, NC
27709-2211

Army Toxic and Hazardous
  Materials Agency
Attn: DRXTH-TE-D   1
Aberdeen Proving Ground, MD
21005

Air Force Armament Division
Attn: AFATL/MNE   1
       AFATL/MNW   1
       AFATL/DLODL   1
       Library   1
Eglin Air Force Base, FL 32542-6009

Air Force Office of Scientific
  Research
Attn: T. Matusko   1
Bolling Air Force Base
Washington, DC 20332

Lawrence Livermore National
  Laboratory
Attn: Technical Library   1
       M. Finger   1
       C. Tarver   1
       E. Lee   1
P.O. Box 808
Livermore, CA 94550

Los Alamos National Laboratory
Attn: Library   1
       J. Repa   1
       P. Howe   1
       P. Tang   1
       A. Bowman   1
       C. Forest   1
P.O. Box 1663
Los Alamos, NM 87545

Sandia National Laboratories
Attn: Technical Library   1
       M Baer   1
Albuquerque, NM 87185

Sandia National Laboratory
Attn: Technical Library   1
P.O. Box 969
Livermore, CA 94550-0096

Argonne National Laboratory
Attn: Records Control   1
       Richard Anderson   1
       Technical Library   1
9700 South Cass Avenue
       Argonne, IL 60439

The Johns Hopkins University
Applied Physics Laboratory
Chemical Propulsion Information
  Agency
Attn: T. W. Christian   1
Johns Hopkins Road
Laurel, MD 20707

The Johns Hopkins University
Applied Physics Laboratory
Attn: Technical Library   1
Johns Hopkins Road
Laurel, MD 20707

New Mexico Institute of Mining
  Technology
Attn: Code TERA (T. Joyner) 1
       Code CETR (P. Persson) 1
Socorro, NM 87801

Applied Research Laboratory
Pennsylvania State University
Attn: Library   1
       E. Liska   1
P.O. Box 30, University Park
State College, PA 16801

DISTRIBUTION (Cont.)

Copies

Internal Distribution:

D4		1
E231		2
E232		3
G22	(W. Holt)	1
G22	(M. Vittoria)	1
R		1
R10		1
R101	(L. Roslund)	1
R10A		1
R10C	(R. Doherty)	1
R12		1
R12	(D. Betancourt)	1
R12	(D. Chung)	1
R12	(L. Hudson III)	10
R12	(L. Mensi)	1
R12	(P. Walter)	1
R13		1
R13	(R. Guirguis)	1
R13	(P. Miller)	1
R13	(K. Kim)	2
R13	(C. Coffey)	1
R13	(H. Sandusky)	1
R13	(D. Tasker)	1
R13	(E. Lemar)	1
R13	(J. Forbes)	1
R13	(R. Bernecker)	1
R13	(G. Sutherland)	1
R14	(H. Mair)	1
R44	(C. Hsieh)	1
U12	(W. Hinckley)	1
U12	(W. Walker)	1

REPORT DOCUMENTATION PAGE

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE 11 September 1990	3. REPORT TYPE AND DATES COVERED Final/May 1990-June 1990

4. TITLE AND SUBTITLE
Program to Produce Tabulated Data Set Describing NSWC Burn Model for Hydrodynamic Computations

5. FUNDING NUMBERS
PE-62314N   PR-RJ14J12
TA-NS3A   WU-R4TBD

6. AUTHOR(S)
Lewis C. Hudson III and Kibong Kim.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Surface Warfare Center
10901 New Hampshire Avenue
Silver Spring MD 20903-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

NAVSWC TR 90-364

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Office of Naval Technology
Washington, DC 20362-5105

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

The concepts of the NSWC Burn Model are put into a general purpose program which runs the one-dimensional hollow sphere pore collapse model for a given input pressure. The general purpose program is run for a series of pressures and the results, at specified intervals, are saved. Other programs which normalize and create FORTRAN statements for inclusion of the tabulated data set into any one-, two-, or three-dimensional hydrocode are presented. The result is an indirect way of describing the NSWC Burn Model via a tabulated data set for use in hydroynamic computations.

14. SUBJECT TERMS
NSWC Kim Burn Model   Composite explosives
FORTRAN   Hydrocode
PBX 9404

15. NUMBER OF PAGES
43

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAR

GENERAL INSTRUCTIONS FOR COMPLETING SF 298

The Report Documentation Page (RDP) is used in announcing and cataloging reports. It is important that this information be consistent with the rest of the report, particularly the cover and its title page. Instructions for filling in each block of the form follow. It is important to *stay within the lines* to meet *optical scanning requirements*.

Block 1. Agency Use Only *(Leave blank)*.

Block 2. Report Date. Full publication date including day, month, and year, if available (e.g. 1 Jan 88). Must cite at least the year.

Block 3. Type of Report and Dates Covered. State whether report is interim, final, etc. If applicable, enter inclusive report dates (e.g. 10 Jun 87 - 30 Jun 88).

Block 4. Title and Subtitle. A title is taken from the part of the report that provides the most meaningful and complete information. When a report is prepared in more than one volume, repeat the primary title, add volume number, and include subtitle for the specific volume. On classified documents enter the title classification in parentheses.

Block 5. Funding Numbers. To include contract and grant numbers; may include program element number(s), project number(s), task number(s), and work unit number(s). Use the following labels:

```
C   -  Contract       PR   -  Project
G   -  Grant          TA   -  Task
PE  -  Program        WU   -  Work Unit
       Element                Accession No.
```

BLOCK 6. Author(s). Name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. If editor or compiler, this should follow the name(s).

Block 7. Performing Organization Name(s) and Address(es). Self-explanatory.

Block 8. Performing Organization Report Number. Enter the unique alphanumeric report number(s) assigned by the organization performing the report.

Block 9. Sponsoring/Monitoring Agency Name(s) and Address(es). Self-explanatory.

Block 10. Sponsoring/Monitoring Agency Report Number. *(If Known)*

Block 11. Supplementary Notes. Enter information not included elsewhere such as: Prepared in cooperation with ..; Trans. of...; To be published in.... When a report is revised, include a statement whether the new report supersedes or supplements the older report.

Block 12a. Distribution/Availability Statement. Denotes public availability or limitations. Cite any availability to the public. Enter additional limitations or special markings in all capitals (e.g. NOFORN, REL, ITAR).

```
DOD   -  See DoDD 5230.24, "Distribution
         Statements on Technical Documents."
DOE   -  See authorities.
NASA  -  See Handbook NHB 2200.2
NTIS  -  Leave blank.
```

Block 12b. Distribution Code.

```
DOD   -  Leave blank.
DOE   -  Enter DOE distribution categories
         from the Standard Distribution for
         Unclassified Scientific and Technical
         Reports.
NASA  -  Leave blank.
NTIS  -  Leave blank.
```

Block 13. Abstract. Include a brief *(Maximum 200 words)* factual summary of the most significant information contained in the report.

Block 14. Subject Terms. Keywords or phrases identifying major subjects in the report.

Block 15. Number of Pages. Enter the total number of pages.

Block 16. Price Code. Enter appropriate price code *(NTIS only)*

Blocks 17.-19. Security Classifications. Self-explanatory. Enter U.S. Security Classification in accordance with U.S. Security Regulations (i.e., UNCLASSIFIED). If form contains classified information, stamp classification on the top and bottom of the page.

Block 20. Limitation of Abstract. This block must be completed to assign a limitation to the abstract. Enter either UL (unlimited) or SAR (same as report). An entry in this block is necessary if the abstract is to be limited. If blank, the abstract is assumed to be unlimited.